Teaching Introductory Computer Science Classes by Utilizing Video Games

Alexander Clark Gunness

Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

28 April 2014
Thirty Second Senior Seminar Conference, Morris

Outline

Outline

Why We Care

- Teaching is difficult
- 40% of students leave in introductory courses
- Alternate methods

ACM Guidelines

- Baseline
- Basic problem solving
- Syntax
- Recursion or non-recursive iteration

ACM Guidelines (cont.)

- No concensus
  - Different devices
- Vague guidelines
- Video games

Difficulties of Recursion

```
public void R1(int x){
  if (x > 0){
    System.out.print(x);
    R1(x - 1);
  }
}
```

Modified from [5]

R1(3) prints 321

```
public void R2(int x){
  if (x > 0){
    R2(x - 1);
    System.out.print(x);
  }
}
```

Modified from [5]

R2(3) prints 123

Difficulties of Recursion (cont.)

```
public void I1(int x){
   for(int i = x; i > 0; i--){
      System.out.print(i);
   }
}
```

I1(3) prints 321

Why Video Games

- Inherent learning
- Motivating
- Computer science style thinking

Outline

Designing an Educational Game

- Still needs to be a fun game
- Learning topic must be incorporated
- Two guidelines

Successful Game Guidelines

- Short, medium, and long-term goals
- Decision-making
- Immediate and specific feedback
- Complex reward system
- Long tasks are usually broken into shorter ones
- Master a specific task before progressing
- Multiple correct solutions, obviously wrong methods

Applied Behavior Analysis (ABA)

- Related to immediate and specific feedback
- ABA basics
- Issues with ABA

ABA Guidelines

- Three Steps
  - Defining/measuring desired behavior
  - Recording/analyzing behavior
  - Feedback
- Adaptation

Outline

Circuitry

- Goal is to reach the exit
- Solve logic gates to progress

Solving Problems

- Toggle inputs
- Constant feedback on correctness

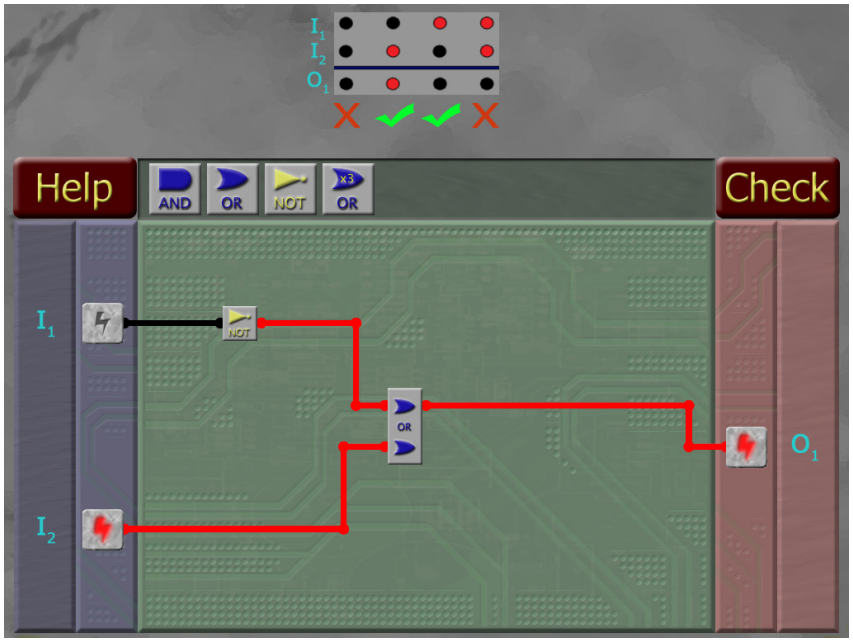


Image from [4]

Pilot Test

- 9 from target group
- Pre-test
  - Eight got 0%
  - One got 100%
  - 0.22/2.00 mean score
- Playing the game
  - 75 minute cap of play
  - Enjoyable
- Post-test
  - Four still got 0%
  - 0.72/2.00 mean score

Future Work

- Larger Study
- Add a plot (for motivation)
- More variety
- Tutorial
- Context-sensitive tips

Circuitry Conclusions

- Successful Games
  - Long Term Goal
  - Feedback
  - Difficulty progression
- ABA
  - Knew what behavior was desired
  - Better feedback (future work)

Recursion with Cargo-Bot

- Adapted commerical game
- Primative language
- Goal is to transform the start state to the end state

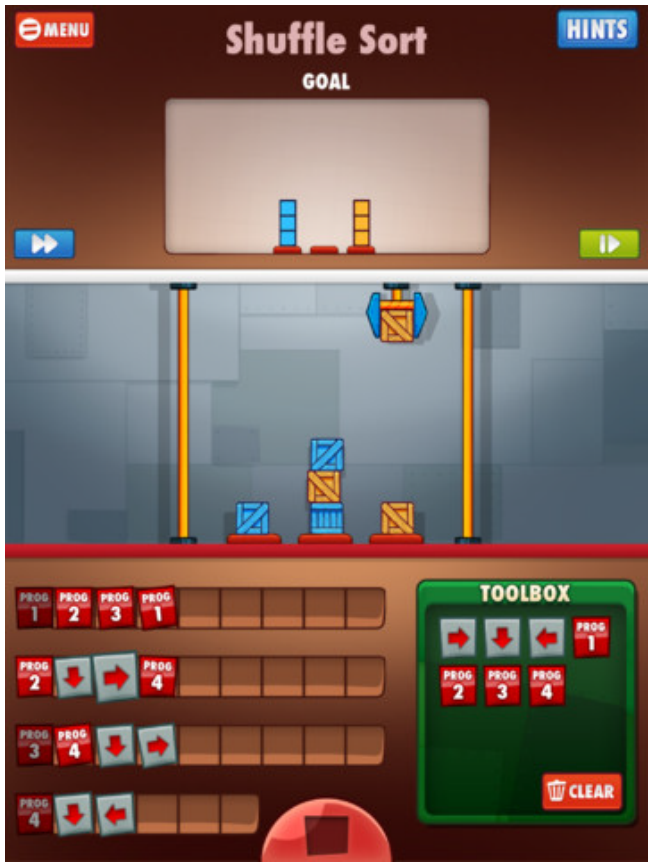


Image from [5]

Why Cargo-Bot?

Exemplifies recursion

- Self-referencing
- has a base case
- Each step progresses to base case

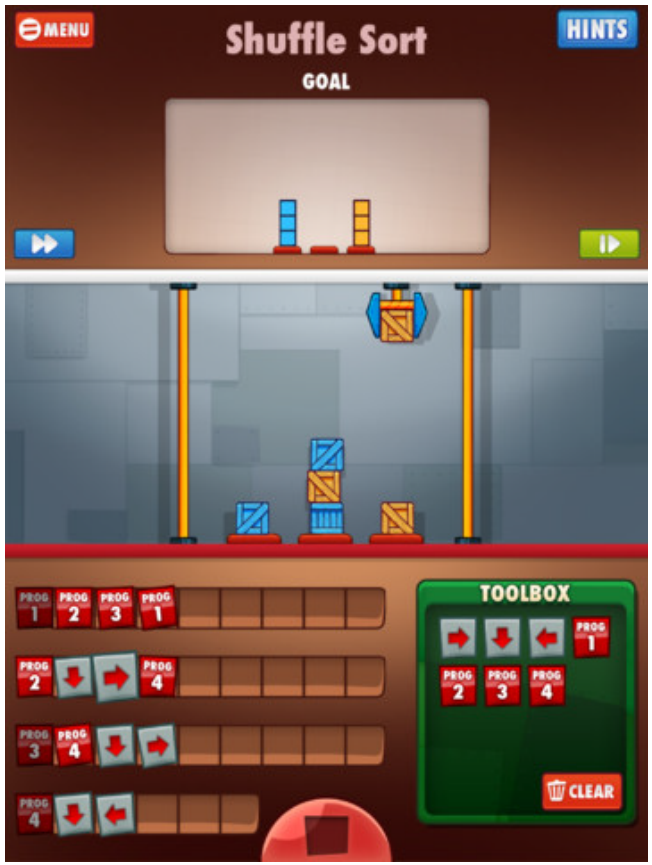


Image from [5]

Test Structure

Cargo-Bot

Control Group

- Day 1
  - 20 Minute pre-test
  - 50 Minute lecture
  - 15 Minute mid-test
- Day 2
  - 90 Minute playing Cargo-Bot
  - 20 Minute post-test

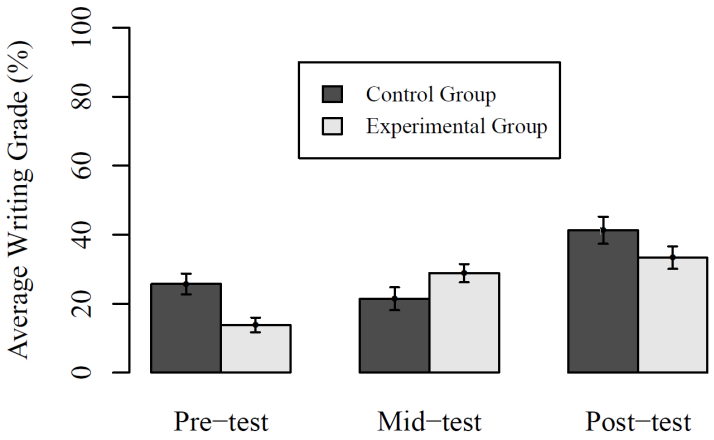
Experimental Group

- Day 1
  - 20 Minute pre-test
  - 90 Minute playing Cargo-Bot
- Day 2
  - 15 Minute mid-test
  - 50 Minute lecture
  - 20 Minute post-test

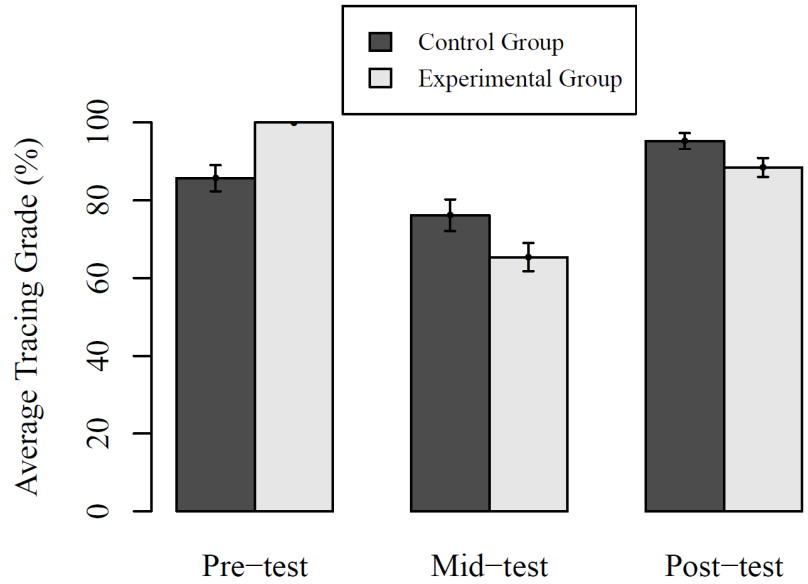
Test Structure (cont.)

- Lecture
  - Recursion
  - Program Stack
  - Recursion examples
- Tests
  - Writing a recursive function
  - Understanding an existing recursive function

Test Results



Scores for Writing Recursive Functions, from [5]



Scores for Reading Recursive Functions, from [5]

Future Work

- Change the way the game carries out procedures

Cargo-Bot Conclusions

- Successful Games
  - Feedback
  - Went against difficulty progression
- ABA
  - Feedback

Recursion with EleMental

Three Levels

- Level 1
  - “Hello World” program
  - Depth-first search (DFS) through a tree manually
- Level 2
  - Scaffolding code
  - Write left-hand side of DFS
- Level 3
  - Write both sides
  - Instruction on program stack
  - Telephone example

```
public void depthFirstSearch(Node node) {
    Thought.moveTo(node);
    // Check for Base Case
    if ((node.returnRight() == null)
        && (node.returnLeft() == null)) {
        return;
    } else { // Recursive calls
        // Travel to node's right child
        if (node.returnRight() != null) {
            depthFirstSearch(node.returnRight());
            Thought.moveTo(node);
        }
        // Travel to node's left child
        if ([YOUR_CODE] != null) {
            [YOUR_CODE]
        }
    }
    return;
}
```

Test Structure

- Pre-test
- Playtime (playtime)
- Post-test
- 43 participants (all taking/took UNCC's "Data Structures and Algorithms" class)
- 16 took pre-test

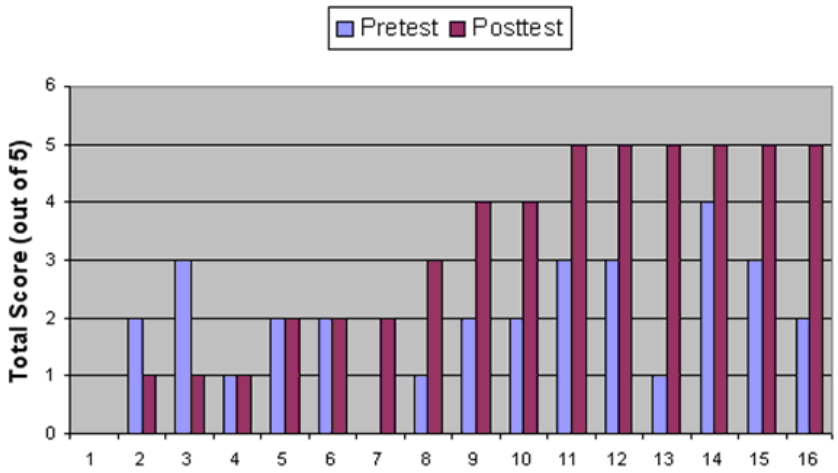


Image from [1]

Future Work

- Show advantages of recursion (replace the “Hello World” part)
- Show what compilable code does
- Improve the telephone metaphor
- Experience point system

EleMental Conclusions

- Successful Games
  - Feedback
  - Difficulty progression
  - Went against having multiple correct solutions
- ABA
  - Desired behavior
  - Feedback (future work)
- Other
  - No significant correlation between time and score

Outline

Overall Conclusions

- None of the examples exemplified step 2 of ABA
- None of the examples had adaptable steps 1-3
- Still unknown if video games have a place

Overall Future Work

- Better research
- Larger sample sizes
- Adhere to the two guidelines

Questions?

References

A. Chaffin, K. Doran, D. Hicks, and T. Barnes.

Experimental evaluation of teaching recursion in a video game.
In *Proceedings of the 2009 ACM SIGGRAPH Symposium on Video Games*, Sandbox '09, pages 79–86, New York, NY, USA, 2009. ACM.

C. Linehan, B. Kirman, S. Lawson, and G. Chan.

Practical, appropriate, empirically-validated guidelines for designing educational games.
In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1979–1988, New York, NY, USA, 2011. ACM.

M. Sahami, A. Danyluk, S. Fincher, K. Fisher, D. Grossman, E. Hawthorne, R. Katz, R. LeBlanc, D. Reed, S. Roach, E. Caudros-Vargas, R. Dodge, R. France, A. Kumar, B. Robinson, R. Seker, and A. Thompson.
Introductory courses.
In *Computer Science Curricula 2013*, Curriculum Guidelines for Undergraduate Degree Programs in Computer Science, pages 39–45. ACM and IEEE, 2013.

V. Srinivasan, K. Butler-Purry, and S. Pedersen.

Using video games to enhance learning in digital systems.
In *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*, Future Play '08, pages 196–199, New York, NY, USA, 2008. ACM.

J. Tessler, B. Beth, and C. Lin.

Using cargo-bot to provide contextualized learning of recursion.
In *Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research*, ICER '13, pages 161–168, New York, NY, USA, 2013. ACM.